**Source: Samsung Electronics Co., Ltd., Vivo, Tencent**

**Title: [FS\_AI4Media] Basic workflow for distributed/federated learning**

**Agenda Item: 9.7**

**Document for: Agreement**

# 1 Introduction

S4-230156r01 is a merge of S4-230151 and S4-230156.

This contribution proposes a basic workflow for the distributed/federated learning basic architecture in PD v0.5.

# 2 Basic architecture and workflow

5.2.3 Distributed/federated learning

5.2.3.1 Basic architectures



**Figure 5.2.3.1-1: Basic architecture for distributed/federated learning between the network and multiple UEs**

Figure 5.2.3-1 shows a simple basic architecture for distributed/federated learning between the network and UE(s), as described in scenario 3) of clause 5.2.

In the network:

* A federated learning engine receives a partially trained model from the AI model repository, that is passed to the AI model delivery function for delivery to multiple UEs via the 5GS.
* Training results data from multiple UEs is also received by the federated learning engine via the 5GS, which is then aggregated for the continuous training of the global model.
* Updates to the global model (e.g. in terms of topology or weights) are delivered to the UEs during the learning process.

In the UE(s):

* AI model data is received by an AI model access function via the 5GS, which then passes the data to the AI training engine.
* An AI training engine in the UE trains the AI model using local device data as the training input.
* Training results (e.g. in the form of updated weights) are delivered to the network via the training results delivery function.

5.2.3.2 Basic workflows

Figure 5.2.3.2-1 shows a basic workflow for distributed/federated learning with training in the UE, the results of which are aggregated in the network. Steps for the procedures shown are described below.



**Figure 5.2.3.2-1: Basic workflow for distributed/federated learning between a UE and the network**

During the initialization and establishment step, it is assumed that information related to the required features and detailed configurations are exchanged and negotiated between the network and UE. Information may include those related to UE device and network capabilities, AI/ML service information (e.g. service requirements, AI/ML model descriptions), and delivery methods. Such information may be used for the selection of a suitable partially trained AI/ML model for the service.

1. The *UE Application* and *Network Application* communicate to trigger distributed/federated learning, using the information from the initialization and establishment step.
2. A partially trained AI model is selected between the *UE Application* and *Network Application*.
3. The *Network Application* identifies the selected partially trained AI model in the *AI model Repository/Provider*.
4. The *AI Model Access Function* establishes an AI model delivery session with the *AI Model Delivery Function*.
5. The *AI Model Access Function* receives the partially trained AI model.
6. The *AI Model Access Function* passes the partially trained AI/ML model to the *AI model Training Engine* in the UE.
7. The *Data Source* passes the training input data to the *AI model Training Engine.*
8. The *AI Model Training Engine* performs AI training.
9. A training result delivery session is established between the *Training Result Delivery Function* and the *Federated Learning Engine*.
10. The *Federated Learning Engine* receives training results data from the UE.
11. The *Federated Learning Engine* performs training aggregation of training results from multiple UEs, and updates the partially trained AI model.
12. The updated partially trained AI model is delivered to the UE as from step 5.

# 3 Proposal

We propose to include the basic workflow for distributed/federated learning corresponding to the basic architecture into the next version of the permanent document.