**3GPP TSG-SA5 Meeting #155 *S5-243095***

**Jeju, South Korea, 27 - 31 May 2024** revision of S5-2412805

**Source: CATT**

**Title: pCR TR 28.9xx Add use case on ML model distributed training**

**Document for: Approval**

**Agenda Item: 6.19.1**

# 1 Decision/action requested

***Approval***

# 2 References

[1] 3GPP TR 28.908 V18.0.0: " Study on Artificial Intelligence/Machine Learning (AI/ML)

management "

[2] SP-231780: "New SID: Study on AI/ML management - phase 2 "

[3] S5-241667-pCR TR 28.908 Add use case on ML model distributed training

# 3 Rationale

It is proposed to add a new use case on ML entity distributed training.

# 4 Detailed proposal

This contribution proposes to make the following changes in [1].

|  |
| --- |
| **1st change** |

### 5.1.x ML model distributed training

#### 5.1.x.1 Description

Distributed training is a model training paradigm that involves spreading training workload across multiple training functions, to accelerate the training process and/or reduce the required computational resources. Distributed training can be used for traditional machine learning models, as well as for large models and compute demanding tasks as deep learning.

In 5GS, the ML training function may be located within the management system or the NF (e.g. gNB or NWDAF), which could be considered as a worker node for training, and each node has different computing resources and storage capacity based on physical infrastructure such as CPU/GPU/DPU, memory, storage, and network bandwidth. In order to obtain load balance between nodes and maximize the efficiency of resource utilization, the training workload is necessary to split up and shared among multiple training functions according to the actual situation of nodes. Thus, aspects of distributed learning needs to be supported in the network management systems.

NOTE: Federated learning is applied to data-distributed training, which focus on address critical issues such as data privacy, data security and data access rights, etc., and training on its own subset of the data.

#### 5.1.x.2 Use cases

#### 5.1.x.2.1 ML model distributed training

ML model parallel is widely-used in distributed training techniques. In 5GS, the MnS consumer may expect to speed up the training process or train the ML model efficiently under certain conditions (e.g. the size of the model may be too large for a single training function), it may require such MnS consumer to transfer the necessary information (e.g., the expected model size, time requirements, or the required computing resources etc.) which could help the MLT MnS producer to determine the training resources required to meet workload performance targets. Moreover, in order to obtain load balance between nodes and maximize the efficiency of resource utilization, the MnS consumer may specify which devices related with ML training functions could be involved in the model training process.

When receiving an ML training request, the MLT MnS producer may evaluate whether distributed training is needed according to the training requirements provided by the ML training consumer, and determine appropriate training function(s) which need to participate in the ML model training.

NOTE: How to split the ML model and synchronize the parameters in different training function depends on the distributed algorithm which are proprietary and not in scope for standardization.

#### 5.1.x.3 Potential requirements

**REQ-ML\_DIST-TRNG-01** The MLT MnS producer should have a capability allowing the authorized consumer to provide requirements for ML distributed training.

**REQ-ML\_DIST-TRNG-02** The MLT MnS producer should have a capability allowing the authorized consumer to provide requirements for selecting appropriate training functions for ML distributed training.

#### 5.1.x.4 Possible solutions

TBD

#### 5.1.x.5 Evaluation

TBD

###

|  |
| --- |
| **End of change** |