**3GPP TSG- Meeting #S5-241943**

**Changsha, CN,15 – Revision of S5-241539**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.2* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** | **0123** | **rev** | **1** | **Current version:** |  |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network | **X** | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Rel-18 CR TS 28.105 correction of ML testing procedure | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Ericsson LG Co | | | | | | | | | |
| ***Source to TSG:*** | S5 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** |  | | | | |  | ***Date:*** | | |  |
|  |  | | | |  | |  | | |  |
| ***Category:*** | F |  | | | | | ***Release:*** | | |  |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18) Rel-19 (Release 19)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | In current definition of workflow, validation is a part of training phase and is performed by MnS training producer before testing and deployment.  In addition, the statement about testing performed by a third party is not relevant to standardization. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Correct the text by including validation in the training phase. Remove non-relevant information about third party actor. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Information will not be aligned in different part of document. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.1, 6.2a.1.2.1, 6.2a.1.2.2, 6.2a.3.1, 6.2a.3.2.1, 6.2a.3.2.2, 6.2a.3.2.3 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

***First change***

# 6 AI/ML management use cases and requirements

## 6.1 General

Each operational step in the workflow (see clause 5.0) is supported by one or more AI/ML management capabilities as depicted below for each of the operational phases.

**Management capabilities for ML training**

**- ML training management**: allowing the MnS consumer to request the ML training, consume and control the producer-initiated training, and manage the ML training/re-training process. The training management capability may include training performance management and setting a policy for the producer-initiated ML training.

**- ML validation**: ML training capability also includes validation to evaluate the performance of the ML entity when performing on the validation data, and to identify the variance of the performance on the training and validation data. If the variance is not acceptable, the ML entity would need to be tuned (re-trained) before being made available for the next step in the operational workflow (e.g., ML entity testing).

**- ML testing management**: allowing the MnS consumer to request the ML entity testing, and to receive the testing results for a trained ML entity. It may also include capabilities for selecting the specific performance metrics to be used or reported by the ML testing function. MnS consumer may also be allowed to trigger ML re-training based on the ML entity testing performance results.

**Management capabilities for ML emulation phase:**

* **AI/ML inference emulation:** a capability allowing an MnS consumer to request an ML inference emulation for a specific ML entity or entities (after the training, validation, and testing) to evaluate the inference performance in an emulation environment prior to applying it to the target network or system.

**Management capabilities for ML entity deployment phase:**

**- ML entity loading management**: allowing the MnS consumer to trigger, control and/or monitor the ML entity loading process.

**Management capabilities for AI/ML inference phase:**

**- AI/ML inference management:** allowing an MnS consumer to control the inference, i.e., activate/deactivate the inference function and/or ML entity/entities, configure the allowed ranges of the inference output parameters. The capabilities also allow the MnS consumer to monitor and evaluate the inference performance and when needed trigger an update of an ML entity or an AI/ML inference function.

The use cases and corresponding requirements for AI/ML management capabilities are specified in the following clauses for each phase of the operational workflow.

***Next change***

#### 6.2a.1.2 Use cases

##### 6.2a.1.2.1 ML training requested by consumer

The ML training capabilities are provided by an ML training MnS producer to one or more consumer(s).

A black background with green lines

Description automatically generated

Figure 6.2a.1.2.1-1: ML training requested by ML training MnS consumer

The ML training may be triggered by the request(s) from one or more ML training MnS consumer(s). The consumer may be for example a network function, a management function, an operator, or another functional differentiation. Figure 6.2a.1.2.1-1 highlights the high-level overview of the process and the relevant sequence.

To trigger an initial ML training, the MnS consumer needs to specify in the ML training request the inference type which indicates the function or purpose of the ML entity, e.g. CoverageProblemAnalysis [see TS 28.104 [2]]. The ML training MnS producer can perform the initial training according to the designated inference type. To trigger an ML re-training, the MnS consumer needs to specify in the ML training request the identifier of the ML entity to be re-trained.

The consumer may provide the data source(s) that contain(s) the training data which are considered as inputs candidates for training. To obtain the valid training outcomes, consumers may also designate their requirements for model performance (e.g. accuracy, etc) in the training request.

The performance of the ML entity depends on the degree of commonality between the distribution of the data used for training and the distribution of the data used for inference. As time progresses, the distribution of the input data used for inference might change as compared to the distribution of the data used for training. In such a scenario, the performance of the ML entity degrades over time. The ML training MnS producer may re-train the ML model associated to the entity if the inference performance of the ML entity falls below a certain threshold, which needs to be configurable by the MnS consumer.

Following the ML training request by the M training MnS consumer, the ML training MnS producer provides a response to the consumer indicating whether the request was accepted.

If the request is accepted, the ML training MnS producer decides when to start the ML training with consideration of the request(s) from the consumer(s). Once the training is decided, the producer performs the following:

- selects the training data, with consideration of the consumer provided candidate training data. Since the training data directly influences the algorithm and performance of the trained ML entity, the ML training MnS producer may examine the consumer's provided training data and decide to select none, some or all of them. In addition, the ML training MnS producer may select some other training data that are available;

- trains the ML model using the selected training data;

- validate the trained model using validation set of the training data;

- provides the training results (including the identifier of the ML entity generated from the initially trained ML model or the version number of the ML entity associated with the re-trained model, training performance results, etc.) to the ML training MnS consumer(s).

***Next change***

### 6.2a.3 ML testing

#### 6.2a.3.1 Description

During ML training phase, after the training and validation, the ML entity needs to be tested to evaluate the performance of the ML entity when it conducts inference using the testing data.

If the testing performance is not acceptable or does not meet the pre-defined requirements, the consumer may request the ML training producer to re-train the ML model with specific training data and/or performance requirements.

#### 6.2a.3.2 Use cases

##### 6.2a.3.2.1 Consumer-requested ML entity testing

After receiving an ML training report about a trained ML entity from the ML training MnS producer, the consumer may request the ML testing MnS producer to test the ML entity before applying it to the target inference function.

The ML testing is to conduct inference on the tested ML entity using the testing data as inference inputs and produce the inference output for each testing dataset example.

The ML testing MnS producer may be the same as or different from the ML training MnS producer.

After completing the ML testing, the ML testing MnS producer provides the testing report indicating the success or failure of the ML testing to the consumer. For a successful ML testing, the testing report contains the testing results, i.e., the inference output for each testing dataset example.

##### 6.2a.3.2.2 Producer-initiated ML entity testing

The ML entity testing may also be initiated by the MnS producer, after the ML entity is trained and validated. A consumer (e.g., an operator) may still need to define the policies (e.g., allowed time window, maximum number of testing iterations, etc.) for the testing of a given ML entity. The consumer may pre-define performance requirements for the ML entity testing and allow the MnS producer to decide on whether re-training/validation need to be triggered. Re-training may be triggered by the testing MnS producer itself based on the performance requirements supplied by the MnS consumer.

##### 6.2a.3.2.3 Joint testing of multiple ML entities

A group of ML entities may work in a coordinated manner for complex use cases.

The group of ML entities is generated by the ML training function. The group, including all contained ML entities, needs to be tested. After the ML testing of the group, the MnS producer provides the testing results to the consumer.

NOTE: This use case is about the ML entities testing during the training phase and is irrelevant to the testing cases that the ML entities have been deployed.

***End of changes***