**3GPP TSG-SA5 Meeting #145-e *S5-225553rev1***

e-meeting, 15 - 24 August 2022

|  |
| --- |
| *CR-Form-v12.1* |
| **CHANGE REQUEST** |
|  |
|  | **28.105** | **CR** | **0005** | **rev** | **-** | **Current version:** | **17.0.0** |  |
|  |
| *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-17 CR TS 28.105 Clarifications and corrections of Use cases  |
|  |  |
| ***Source to WG:*** | NEC, Intel |
| ***Source to TSG:*** | S5  |
|  |  |
| ***Work item code:*** | eMDAS |  | ***Date:*** | 2022-08-01 |
|  |  |  |  |  |
| ***Category:*** | **F** |  | ***Release:*** | Rel-17 |
|  | *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 canbe 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-15 (Release 15)Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)* |
|  |  |
| ***Reason for change:*** | Number of text instances in the Use case clause either needed corrections and/or clarifications.  |
|  |  |
| ***Summary of change:*** | Some further clarifications and corrections are made for the following Use cases in clause 6.2;* AI/ML training requested by consumer
* AI/ML training initiated by producer
* Selecting AI/ML models and AI/ML-enabled Functions
* Managing AI/ML Training Processes, and
* Handling errors in data and ML decisions
 |
|  |  |
| ***Consequences if not approved:*** | Ambiguties leading to misunderstanding of the use cases which may result in misunderstanding of the relevant specifications and consequently in wrong and non-interoperable implementations.  |
|  |  |
| ***Clauses affected:*** | 6.2.2.1, 6.2.2.2, 6.2.2.3, 6.2.2.4, 6.2.2.5 |
|  |  |
|  | **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:*** |  |

|  |
| --- |
| **1st Change** |

### 6.2.2 Use cases

#### 6.2.2.1 AI/ML training requested by consumer

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



Figure 6.2.2.1-1: AI/ML training requested by AIMLT MnS consumer

The AI/ML training may be triggered by the request(s) from one or more AIMLT MnS consumer(s). The consumer may be for example a network function, a management function, an operator, or another functional differentiation. To trigger an AI/ML training, the AIMLT MnS consumer requests the AIMLT MnS producer to train the AI/ML model or AI/ML enabled function. In the AI/ML training request, the consumer should specify the inference type which indicates the function or purpose of the AI/ML Entity, e.g. CoverageProblemAnalysis. The AIMLT MnS producer can perform the training according to the designated inference type. 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 AIMLT MnS producer provides a response to the consumer indicating whether the request was accepted.

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

- 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 AI/ML Entity, the AIMLT MnS producer may examine the consumer's provided training data and decide to select none, some or all of them. In addition, the AIMLT MnS producer may select some other training data that are available;

- trains the AI/ML Entity using the selected training data; and

- provides the training results (including the location of the trained AI/ML Entity, etc.) to the AIMLT MnS consumer(s).

#### 6.2.2.2 AI/ML training initiated by producer

The AI/ML training may be initiated by the AIMLT MnS producer, for instance as a result of performance evaluation of the AI/ML model, based on feedback or new training data received from the consumer, or when new training data which are not from the consumer describing the new network status/events become available.

When the AIMLT MnS producer decides to start the AI/ML training, the producer performs the followings:

- selects the training data;

- trains the AI/ML Entity using the selected training data; and

- provides the training results (including the location of the trained AI/ML Entity, etc.) to the AIMLT MnS consumer(s) who have subscribed to receive the AI/ML training results.

#### 6.2.2.3 Selecting AI/ML models and AI/ML-enabled Functions

For a given machine learning-based use case, different entities that apply the respective ML model or AI/ML enabled function may have different inference requirements and capabilities. For example, one consumer with specific responsibility and wish to have an AI/ML enabled function trained for city central business district where mobile users move at speeds not exceeding 30 km/hr. On the other hand, another consumer for the same use case may support a rural environment and as such wish to have a model fitting that environment. The different consumers need to know the available versions of AI/ML enabled functions and to select the appropriate AI/ML enabled function for their respective conditions.

Besides there is no guarantee that the available AI/ML enabled functions have been trained according to the characteristics that the consumers expect. As such the consumers need to know the conditions for which the models or AI/ML enabled functions have been trained to then enable them to select the models that are best fit to their conditions and needs.

The models that have been trained may differ in terms of complexity and performance. For example, a generic comprehensive and complex model may have been trained in a cloud-like environment but when such a model cannot be used in the gNB and instead, a less complex model, trained as a derivative of this generic model, could be a better candidate. Moreover, multiple less complex models could be trained with different level of complexity and performance which would then allow different relevant models to be delivered to different network functions depending on operating conditions and performance requirements. The network functions need to know the alternative models available and interactively request and replace them when needed and depending on the observed inference‑related constraints and performance requirements.

#### 6.2.2.4 Managing AI/ML Training Processes

This machine learning capability relates to means for managing and controlling AI/ML training processes.

To achieve the desired outcomes of any machine learning relevant use-case, the AI/ML Model applied for such analytics and decision making, needs to be trained with the appropriate data. The training may be undertaken in managed function or in a management function.

In either case, the network (or the OAM system thereof) not only needs to have the required training capabilities but needs to also have the means to manage the training of the AI/ML models and or AI/ML-enabled functions. The consumers need to be able to interact with the training process, e.g. to suspend or restart the process; and also need to manage and control the requests related to any such training process.

#### 6.2.2.5 Handling errors in data and ML decisions

Traditionally, the machine-learning-enabled Functions (e.g. AIML Entity 1 and AIML Entity) are trained on good quality data, i.e. data that were collected correctly and reflected the real network status to represent the expected context in which the AIML Entity is meant to operate. Good quality data is void of errors, such as:

- Imprecise measurements, with added noise (such as RSRP, SINR, or QoE estimations).

- Missing values or entire records, e.g. because of communication link failures.

- Records which are communicated with a significant delay (in case of online measurements).

Without errors, an AIML Entity can depend on a few precise inputs, and don't need to exploit the redundancy present in the training data. However, during inference, the AIML Entity is very likely to come across these inconsistencies. When this happens, the AIML Entity shows high error in the inference outputs, even if redundant and uncorrupted data are available from other sources.

AIML Entity1

AIML Entity2

Network Resources

ML Consumer

p

KPIs

**Error**

p

Figure 6.2.2.5-1: The propagation of erroneous information

As such the system needs to account for errors and inconsistencies in the input data and the consumers of should deal with decisions that are made based on such erroneous and inconsistent data. The system should:

1) enable functions to undertake the training in a way that prepares the AIML Entity s to deal with the errors in the training data, i.e. to identify the errors in the data during training; and

2) enable the MLT MnS consumers to be aware of the possibility of erroneous input data that are used by the ML entity.

|  |
| --- |
| **End of Change** |