**3GPP TSG- Meeting # *d1***

**, , -**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.3* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** |  | **rev** |  | **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-17 CR TS 28.105 Remove attribute inferencetype and replace it with capabilityName | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | , Huawai? | | | | | | | | | |
| ***Source to TSG:*** | SA5 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** |  | | | | |  | ***Date:*** | | |  |
|  |  | | | |  | |  | | |  |
| ***Category:*** |  |  | | | | | ***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-17 (Release 17) Rel-18 (Release 18) Rel-19 (Release 19)  Rel-20 (Release 20)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | In AI/ML Community the term “Inference Type” or “Type Inference” is known to be the process that determine the type of an expression or variable at compile time, associated with the data provided.  In TS 28.105 attribute inferenceType is defined in 7.5.1 as the ” the type of inference that the ML model supports and its allowed values are: the values of the MDA type (see 3GPP TS 28.104 [2]), Analytics ID(s) of NWDAF (see 3GPP TS 23.288 [3]), types of inference for RAN, and vendor's specific extensions, therefore the definition of inference type is different than the one known in AI/ML community”.  The way inferenceType is specified in 3GPP can lead to problems with implementation and cause readability issues. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Rename attribute inferenceType to aIMLInferenceName and correct definition of capabilityName attribute. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | lead to incorrect implementation. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.2.1, 7.4.2.1, 6.2.3 , B.2.1 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | | Forge MR link: <https://forge.3gpp.org/rep/sa5/MnS/-/merge_requests/1103> at commit 31220ecd55c553b7c62df053fbdd7ecfac914101 | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

|  |
| --- |
| **Start of modification** |

## 6.2 ML training

### 6.2.1 Description

In operational environment before the ML entity is deployed to conduct inference, the ML model associated with the ML entity needs to be trained (e.g. by ML training function which may be a separate or an external entity to the AI/ML inference function).

NOTE: In the present document, ML entity training refers to ML model training associated with an ML entity.

The ML Entity is trained by the ML training (MLT) MnS producer, and the training can be triggered by request(s) from one or more MLT MnS consumer(s), or initiated by the MLT MnS producer (e.g. as result of model evaluation).

### 6.2.2 Use cases

#### 6.2.2.1 ML training requested by consumer

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



Figure 6.2.2.1-1: ML training requested by MLT MnS consumer

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

If the request is accepted, the MLT 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 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 ML Entity, the MLT MnS producer may examine the consumer's provided training data and decide to select none, some or all of them. In addition, the MLT MnS producer may select some other training data that are available;

- trains the ML entity using the selected training data;

- provides the training results to the MLT MnS consumer(s).

|  |
| --- |
| **Next modification** |

### 7.4.2 MLEntity <<dataType>>

#### 7.4.2.1 Definition

This data type represents the properties of an ML entity. ML training may be requested for either an ML model or ML entity. ML model is not to be standardized.

For each MLEntity under training, one or more MLTrainingProcess are instantiated.

The MLEntity may contain 3 types of contexts - TrainingContext which is the context under which the MLEntity has been trained, the ExpectedRunTimeContext which is the context where an MLEntity is expected to be applied or/and the RunTimeContext which is the context where the MLmodel or entity is being applied.

#### 7.4.2.2 Attributes

Table 7.4.2.2-1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifyable |
| mLEntityId | M | T | F | F | T |
| aIMLInferenceName | M | T | F | F | T |
| mLEntityVersion | M | T | F | F | T |
| expectedRunTimeContext | O | T | T | F | T |
| trainingContext | CM | T | F | F | T |
| runTimeContext | O | T | F | F | T |

|  |
| --- |
| **Next modification** |

### 6.2.3 Requirements for ML training

Table 6.2.3-1

| Requirement label | Description | Related use case(s) |
| --- | --- | --- |
| **REQ-ML\_TRAIN-FUN-01** | The MLT MnS producer shall have a capability allowing the consumer to request ML training. | ML training requested by consumer (clause 6.2.2.1) |
| **REQ- ML\_TRAIN-FUN-02** | The MLT MnS producer shall have a capability allowing the consumer to specify the data sources containing the candidate training data for ML training. | ML training requested by consumer (clause 6.2.2.1) |
| **REQ- ML\_TRAIN-FUN-03** | The MLT MnS producer shall have a capability allowing the consumer to specify the AI/ML inference nameof the ML model to be trained. | ML training requested by consumer (clause 6.2.2.1) |
| **REQ- ML\_TRAIN-FUN-04** | The MLT MnS producer shall have a capability to provide the training result to the consumer. | ML training requested by consumer (clause 6.2.2.1), /ML training initiated by producer (clause 6.2.2.2) |
| **REQ-ML\_SELECT-01** | 3GPP management system shall have the capability to enable an authorized consumer to discover the characteristics of available models including the contexts under which each of the models was trained. | ML model and ML entity selection (clause 6.2.2.3) |
| **REQ-ML\_SELECT-02** | 3GPP management system shall have the capability to enable an authorized consumer to select an ML model. | ML models and ML entity selection (clause 6.2.2.3) |
| **REQ-ML\_SELECT-03** | The MLT MnS producer shall have the capability to enable an authorized consumer to request for a model to be trained to satisfy the consumer's expectations. | ML training requested by consumer (clause 6.2.2.1), ML model and ML entity selection (clause 6.2.2.3) |
| **REQ-ML\_SELECT-04** | 3GPP management system shall have the capability to enable an authorized consumer to request for information and be informed about the available alternative models of differing complexity and performance. | ML model and ML entity selection (clause 6.2.2.3) |
| **REQ-ML\_SELECT-05** | 3GPP management system shall have the capability to enable an authorized consumer to request one of the known or available alternative models of differing complexity and performance to be used for inference. | ML model and ML entity selection (clause 6.2.2.3) |
| **REQ-ML\_SELECT-06** | The 3GPP management system shall have a capability to provide a selected ML model/entity to the consumer. | ML model and ML entity selection (clause 6.2.2.3) |
| **REQ-ML\_TRAIN- MGT-01** | The MLT MnS producer shall have a capability allowing an authorized consumer to manage and configure one or more requests for the training of specific ML models or ML entities, e.g. to modify the characteristics of the request or to delete a request. | ML training requested by consumer (clause 6.2.2.1),Managing ML Training Processes (clause 6.2.2.4) |
| **REQ-ML\_TRAIN- MGT-02** | The MLT MnS producer shall have a capability allowing an authorized consumer to manage and configure one or more training processes, e.g. to start, suspend or restart the training; or to adjust the training conditions and/or characteristics. | ML training requested by consumer (clause 6.2.2.1),  Managing ML training processes (clause 6.2.2.4) |
| **REQ-ML\_TRAIN- MGT-03** | 3GPP management system shall have the capability to enable an authorized consumer (e.g. the function/entity different from the function that generated a request for ML model/entity training) to request for a report on the outcomes of a specific training instance. | Managing ML training processes (clause 6.2.2.4) |
| **REQ-ML\_TRAIN- MGT-04** | 3GPP management system shall have the capability to enable an authorized consumer to define the reporting characteristics related to a specific training request or training instance. | Managing ML training processes (clause 6.2.2.4) |
| **REQ-ML\_TRAIN- MGT-05** | 3GPP management system shall have the capability to enable the MLT function to report to any authorized consumer about specific ML Training process and/or report about the outcomes of any such ML training process. | Managing ML training processes (clause 6.2.2.4) |
| **REQ-ML\_ERROR-01** | The 3GPP management system shall enable an authorized consumer of data services (e.g. an MLT function) to request from a producer of data services a Value Quality Score of the data, which is the numerical value that represents the dependability/quality of a given observation and measurement type. | Handling errors in data and ML decisions (clause 6.2.2.5) |
| **REQ-ML\_ERROR-02** | The 3GPP management system shall enable an authorized consumer of AI/ML decisions (e.g. a controller) to request ML decision confidence score which is the numerical value that represents the dependability/quality of a given decision generated by an AI/ML-inference function. | Handling errors in data and ML decisions (clause 6.2.2.5) |
| **REQ-ML\_ERROR-03** | The 3GPP management system shall enable a producer of data services (e.g. a gNB) to provide to an authorized consumer (e.g. an MLT function) a Value Quality Score of the data, which is the numerical value that represents the dependability/quality of a given observation and measurement type. | Handling errors in data and ML decisions (clause 6.2.2.5) |
| **REQ-ML\_ERROR-04** | The 3GPP management system shall enable a producer of ML decisions (e.g. an AI/ML inference function) to provide to an authorized consumer of AI/ML decisions (e.g. a controller) an AI/ML decision confidence score which is the numerical value that represents the dependability/quality of a given decision generated by the inference function. | Handling errors in data and ML decisions (clause 6.2.2.5) |

|  |
| --- |
| **Next modification** |

### 7.5.1 Attribute properties

Table 7.5.1-1

| Attribute Name | Documentation and Allowed Values | Properties |
| --- | --- | --- |
| mLEntityId | It identifies the ML entity.  It is unique in each MnS producer.  allowedValues: N/A. | type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| candidateTrainingDataSource | It provides the address(es) of the candidate training data source provided by MnS consumer. The detailed training data format is vendor specific.  allowedValues: N/A. | type: String  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| aIMLInferenceName | It indicates the type of inference that the ML model supports.  allowedValues: the values of the MDA type (see 3GPP TS 28.104 [2]), Analytics ID(s) of NWDAF (see 3GPP TS 23.288 [3]), types of inference for RAN‑intelligence, and vendor's specific extensions. | type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| areConsumerTrainingDataUsed | It indicates whether the consumer provided training data have been used for the ML model training.  allowedValues: ALL, PARTIALLY, NONE. | type: Enum  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| usedConsumerTrainingData | It provides the address(es) where lists of the consumer-provided training data are located, which have been used for the ML model training.  allowedValues: N/A. | type: String  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| trainingRequestRef | It is the DN(s) of the related MLTrainingRequest MOI(s).  allowedValues: DN. | type: DN (see TS 32.156 [13])  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| trainingProcessRef | It is the DN(s) of the related MLTrainingProcess MOI(s) that produced the MLTrainingReport.  allowedValues: DN. | type: DN (see TS 32.156 [13])  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| trainingReportRef | It is the DN of the MLTrainingReport MOI that represents the reports of the ML training.  allowedValues: DN. | Type: DN (see TS 32.156 [13])  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| lastTrainingRef | It is the DN of the MLTrainingReport MOI that represents the reports for the last training of the ML model.  allowedValues: DN. | type: DN (see 3GPP TS 32.156 [13])  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| modelConfidenceIndication | It indicates the average confidence value (in unit of percentage) that the ML model would perform for inference on the data with the same distribution as training data.  Essentially, this is a measure of degree of the convergence of the trained ML model.  allowedValues: { 0..100 }. | type: integer  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| mLEntityList | It describes the list of MLEntity. | type: MLEntity  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| trainingRequestSource | It describes the entity that requested to instantiate the MLTrainingRequest MOI. | type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| requestStatus | It describes the status of a particular ML training request.  allowedValues: NOT\_STARTED, TRAINING\_IN\_PROGRESS, CANCELLING, SUSPENDED, FINISHED, and CANCELLED. | type: Enum  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| mLTrainingProcessId | It identifies the training process.  It is unique in each instantiated process in the MnS producer.  allowedValues: N/A. | type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| priority | It indicates the priority of the training process.  The priority may be used by the ML training to schedule the training processes. Lower value indicates a higher priority.  allowedValues: { 0..65535 }. | type: integer  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: 0  isNullable: False |
| terminationConditions | It indicates the conditions to be considered by the MLtraining MnS producer to terminate a specific training process.  allowedValues: MODEL UPDATED\_IN\_INFERENCE\_FUNCTION, INFERENCE FUNCTION\_TERMINATED, INFERENCE FUNCTION\_UPGRADED, INFERENCE\_CONTEXT\_CHANGED | type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| progressStatus | It indicates the status of the ML training process.  allowedValues: N/A. | type: ProcessMonitor (see TS 28.622 [12])  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| mLEntityVersion | It indicates the version number of the ML entity.  allowedValues: N/A. | type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| performanceRequirements | It indicates the expected performance for a trained ML entity when performing on the training data.  allowedValues: N/A. | type: ModelPerformance  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| modelperformanceTraining | It indicates the performance score of the ML entity when performing on the training data.  allowedValues: N/A. | type: ModelPerformance  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| mLTrainingProcess.progressStatus.progressStateInfo | It provides the following specialization for the "progressStateInfo" attribute of the "ProcessMonitor" data type for the "MLTrainingProcess".  When the ML training is in progress, and the " mLTrainingProcess.progressStatus.status " is equal to "RUNNING", it provides the more detailed progress information.  allowedValues for " mLTrainingProcess.progressStatus.status " = "RUNNING":  - COLLECTING\_DATA  - PREPARING\_TRAINING\_DATA  - TRAINING  The allowed values for " mLTrainingProcess.progressStatus.status " = "CANCELLED" are vendor specific. | Type: String  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| inferenceOutputName | It indicates the name of an inference output of an ML entity.  allowedValues: the name of the MDA output IEs (see 3GPP TS 28.104 [2]), name of analytics output IEs of NWDAF (see TS 23.288 [3]), RAN-intelligence inference output IE name(s), and vendor's specific extensions. | Type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| performanceMetric | It indicates the performance metric used to evaluate the performance of an ML entity, e.g. "accuracy", "precision", "F1 score", etc.  allowedValues: N/A. | Type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| performanceScore | It indicates the performance score (in unit of percentage) of an ML entity when performing inference on a specific data set (Note).  The performance metrics may be different for different kinds of ML models depending on the nature of the model. For instance, for numeric prediction, the metric may be accuracy; for classification, the metric may be a combination of precision and recall, like the "F1 score".  allowedValues: { 0..100 }. | Type: Real  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| cancelRequest | It indicates whether the ML training MnS consumer cancels the ML training request.  Setting this attribute to "TRUE" cancels the ML training request. Cancellation is possible when the requestStatus is the "NOT\_STARTED", " TRAINING\_IN\_PROGRESS", and "SUSPENDED" state. Setting the attribute to "FALSE" has no observable result.  Default value is set to "FALSE".  allowedValues: TRUE, FALSE. | Type: Boolean  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: FALSE  isNullable: False |
| suspendRequest | It indicates whether the ML training MnS consumer suspends the /ML training request.  Setting this attribute to "TRUE" suspends the ML training request. Suspension is possible when the requestStatus is not the "FINISHED" state. Setting the attribute to "FALSE" has no observable result.  Default value is set to "FALSE".  allowedValues: TRUE, FALSE. | Type: Boolean  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: FALSE  isNullable: False |
| cancelProcess | It indicates whether the ML training MnS consumer cancels the ML training process.  Setting this attribute to "TRUE" cancels the ML training request. Cancellation is possible when the " mLTrainingProcess.progressStatus.status" is not the "FINISHED" state. Setting the attribute to "FALSE" has no observable result.  Default value is set to "FALSE".  allowedValues: TRUE, FALSE. | Type: Boolean  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: FALSE  isNullable: False |
| suspendProcess | It indicates whether the ML training MnS consumer suspends the ML training process.  Setting this attribute to "TRUE" suspends the ML training request. Suspension is possible when the " mLTrainingProcess.progressStatus.status" is not the "FINISHED", "CANCELLING" or "CANCELLED" state. Setting the attribute to "FALSE" has no observable result.  Default value is set to "FALSE".  allowedValues: TRUE, FALSE. | Type: Boolean  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: FALSE  isNullable: False |
| inferenceEntityRef | It describes the target entities that will use the ML entity for inference. | Type: DN (see 3GPP TS 32.156 [13])  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| dataProviderRef | It describes the entities that have provided or should provide data needed by the ML entity e.g. for training or inference | Type: DN (see 3GPP TS 32.156 [13])  multiplicity: \*  isOrdered: False  isUnique: True  defaultValue: None  isNullable: False |
| areNewTrainingDataUsed | It indicates whether the other new training data have been used for the ML model training.  allowedValues: TRUE, FALSE. | type: Boolean  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| trainingDataQualityScore | It indicates numerical value that represents the dependability/quality of a given observation and measurement type. The lowest value indicates the lowest level of dependability of the data, i.e. that the data is not usable at all.  allowedValues: { 0..100 }. | Type: Real  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| decisionConfidenceScore | It is the numerical value that represents the dependability/quality of a given decision generated by the AI/ML inference function. The lowest value indicates the lowest level of dependability of the decisions, i.e. that the data is not usable at all.  allowedValues: { 0..100 }. | Type: Real  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| expectedRuntimeContext | This describes the context where an MLEntity is expected to be applied or/and the RunTimeContext which is the context where the MLmodel or entity is being applied.  allowedValues: NA | Type: MLContext  multiplicity: 0..1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| trainingContext | This specify the context under which the MLEntity has been trained.  allowedValues: NA | Type: MLContext  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| runTimeContext | This specifies the context where the MLmodel or entity is being applied.  allowedValues: NA | Type: MLContext  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| capabilityName | It indicates the name of a capability for which an ML entity can generate inference.  allowedValues: N/A. | type: String  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False |
| NOTE: When the performanceScore is to indicate the performance score for ML training, the data set is the training data set. | | |

|  |
| --- |
| **Next modification** |

## B.2.1 OpenAPI document "TS28105\_AiMlNrm.yaml"

<CODE BEGINS>

openapi: 3.0.1

info:

title: AI/ML NRM

version: 17.7.0

description: >-

OAS 3.0.1 specification of the AI/ML NRM

© 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).

All rights reserved.

externalDocs:

description: 3GPP TS 28.105; AI/ML Management

url: http://www.3gpp.org/ftp/Specs/archive/28\_series/28.105/

paths: {}

components:

schemas:

#-------- Definition of types-----------------------------------------------------

MLEntityList:

type: array

items:

$ref: '#/components/schemas/MLEntity'

MLEntity:

type: object

properties:

mLEntityId:

type: string

aIMLInferenceName:

type: string

mLEntityVersion:

type: string

expectedRunTimeContext:

$ref: '#/components/schemas/MLContext'

trainingContext:

$ref: '#/components/schemas/MLContext'

runTimeContext:

$ref: '#/components/schemas/MLContext'

MLContext:

type: object

properties:

inferenceEntityRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

dataProviderRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

RequestStatus:

type: string

enum:

- NOT\_STARTED

- TRAINING\_IN\_PROGRESS

- SUSPENDED

- FINISHED

- CANCELLED

PerformanceRequirements:

type: array

items:

$ref: '#/components/schemas/ModelPerformance'

ModelPerformance:

type: object

properties:

inferenceOutputName:

type: string

performanceMetric:

type: string

performanceScore:

type: number

format: float

decisionConfidenceScore:

type: number

format: float

TrainingProcessMonitor:

description: >-

This data type is the "ProcessMonitor" data type defined in “genericNrm.yaml” with specialisations for usage in the "MLTrainingProcess".

type: object

properties:

mLTrainingProcessId:

type: string

status:

type: string

enum:

- RUNNING

- CANCELLING

- CANCELLED

- SUSPENDED

- FINSHED

progressPercentage:

type: integer

minimum: 0

maximum: 100

progressStateInfo:

type: string

enum:

- COLLECTING\_DATA

- PREPARING\_TRAINING\_DATA

- TRAINING

resultStateInfo:

type: string

#-------- Definition of abstract IOCs --------------------------------------------

#-------- Definition of concrete IOCs --------------------------------------------

SubNetwork-Single:

allOf:

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/Top'

- type: object

properties:

attributes:

$ref: 'TS28623\_GenericNrm.yaml#/components/schemas/SubNetwork-Attr'

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/SubNetwork-ncO'

- type: object

properties:

SubNetwork:

$ref: '#/components/schemas/SubNetwork-Multiple'

ManagedElement:

$ref: '#/components/schemas/ManagedElement-Multiple'

MLTrainingFunction:

$ref: '#/components/schemas/MLTrainingFunction-Multiple'

ManagedElement-Single:

allOf:

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/Top'

- type: object

properties:

attributes:

$ref: 'TS28623\_GenericNrm.yaml#/components/schemas/ManagedElement-Attr'

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/ManagedElement-ncO'

- type: object

properties:

MLTrainingFunction:

$ref: '#/components/schemas/MLTrainingFunction-Multiple'

MLTrainingFunction-Single:

allOf:

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/Top'

- type: object

properties:

attributes:

allOf:

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/ManagedFunction-Attr'

- type: object

properties:

mLEntityList:

$ref: '#/components/schemas/MLEntityList'

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/ManagedFunction-ncO'

- type: object

properties:

MLTrainingRequest:

$ref: '#/components/schemas/MLTrainingRequest-Multiple'

MLTrainingProcess:

$ref: '#/components/schemas/MLTrainingProcess-Multiple'

MLTrainingReport:

$ref: '#/components/schemas/MLTrainingReport-Multiple'

MLTrainingRequest-Single:

allOf:

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/Top'

- type: object

properties:

attributes:

allOf:

- type: object

properties:

mLEntityId:

type: string

candidateTrainingDataSource:

type: array

items:

type: string

trainingDataQualityScore:

type: number

format: float

trainingRequestSource:

oneOf:

- type: string

- $ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

requestStatus:

$ref: '#/components/schemas/RequestStatus'

expectedRuntimeContext:

$ref: '#/components/schemas/MLContext'

performanceRequirements:

$ref: '#/components/schemas/PerformanceRequirements'

cancelRequest:

type: boolean

suspendRequest:

type: boolean

MLTrainingProcess-Single:

allOf:

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/Top'

- type: object

properties:

attributes:

allOf:

- type: object

properties:

mLTrainingProcessId:

type: string

priority:

type: integer

terminationConditions:

type: string

progressStatus:

$ref: '#/components/schemas/TrainingProcessMonitor'

cancelProcess:

type: boolean

suspendProcess:

type: boolean

trainingRequestRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

trainingReportRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

MLTrainingReport-Single:

allOf:

- $ref: 'TS28623\_GenericNrm.yaml#/components/schemas/Top'

- type: object

properties:

attributes:

allOf:

- type: object

properties:

mLEntityId:

type: string

areConsumerTrainingDataUsed:

type: boolean

usedConsumerTrainingData:

type: array

items:

type: string

modelConfidenceIndication:

type: integer

modelPerformanceTraining:

type: array

items:

$ref: '#/components/schemas/ModelPerformance'

areNewTrainingDataUsed:

type: boolean

trainingRequestRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

trainingProcessRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

trainingReportRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

lastTrainingRef:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

#-------- Definition of JSON arrays for name-contained IOCs ----------------------

SubNetwork-Multiple:

type: array

items:

$ref: '#/components/schemas/SubNetwork-Single'

ManagedElement-Multiple:

type: array

items:

$ref: '#/components/schemas/ManagedElement-Single'

MLTrainingFunction-Multiple:

type: array

items:

$ref: '#/components/schemas/MLTrainingFunction-Single'

MLTrainingRequest-Multiple:

type: array

items:

$ref: '#/components/schemas/MLTrainingRequest-Single'

MLTrainingProcess-Multiple:

type: array

items:

$ref: '#/components/schemas/MLTrainingProcess-Single'

MLTrainingReport-Multiple:

type: array

items:

$ref: '#/components/schemas/MLTrainingReport-Single'

#-------- Definitions in TS 28.104 for TS 28.532 ---------------------------------

resources-AiMlNrm:

oneOf:

- $ref: '#/components/schemas/SubNetwork-Single'

- $ref: '#/components/schemas/ManagedElement-Single'

- $ref: '#/components/schemas/MLTrainingFunction-Single'

- $ref: '#/components/schemas/MLTrainingRequest-Single'

- $ref: '#/components/schemas/MLTrainingProcess-Single'

- $ref: '#/components/schemas/MLTrainingReport-Single'

<CODE ENDS>

\*\*\* END OF CHANGE 1 \*\*\*

|  |
| --- |
| **End of modification** |