**3GPP TSG-SA Meeting #106SP-241840**

**10 - 13 December 2024, Madrid, Spain**

**Source: Nokia**

**Title: pCR TR 22.850 AIML Terminology analysis across 3GPP**

**Document for: Approval**

**Agenda Item: 7**

# 1 Decision/action requested

***The group is asked to discuss and agree on the proposal.***

# 2 References

None.

# 3 Rationale

This pCR aims to propose solutions to resolve inconsistencies and misalignments among AI/ML terminologies across 3GPP as identified in SA#105 meeting.

# 4 Detailed proposal

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

# 6 Analysis on AI/ML across 3GPP

# 6 Analysis on AI/ML across 3GPP

## 6.1 General

This clause will identify any potential misalignments and inconsistencies for AI/ML across 3GPP, based on clause 5.

NOTE: Any RAN related aspects are subject to early coordination and feedback from TSG RAN.

## 6.2 AI/ML related terminology

### 6.2.1 Analysis on terminology consistency

This clause identifies any potential misalignments and inconsistencies for AI/ML terminology across 3GPP, based on clause 5.

#### 6.2.1.1 Analysis on ML model

The term 'ML model' has been defined differently by SA WG5, SA WG6 and RAN WG1, as illustrated in Table 6.2.1.1-1.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.1.1-1: Definition of ML model as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | ML model |
| SA WG5 TS 28.105 [9] | A manageable representation of an ML model algorithm.  NOTE 1: An ML model algorithm is a mathematical algorithm through which running a set of input data can generate a set of inference output.  NOTE 2: An ML model algorithm is proprietary and not in scope for standardization and therefore not treated in this specification.  NOTE 3: An ML model may include metadata. Metadata may include e.g. information related to the trained model, and applicable runtime context. |
| SA WG6 TR 23.700-82 [7] | According to TS 28.105 [9], mathematical algorithm that can be "trained" by data and human expert input as examples to replicate a decision an expert would make when provided that same information. |
| RAN WG1 TR 38.843 [3] | A data driven algorithm that applies AI/ML techniques to generate a set of outputs based on a set of inputs. |

The following unified definition for ‘ML model’ is proposed:

**ML model:** A mathematical algorithm that applies ML techniques to generate a set of outputs based on a set of inputs. It may include metadata which consists of, e.g., information related to the model, and applicable runtime context.

NOTE:An ML model can be managed, stored, and transferred as artifacts, which may be containers, images, or proprietary file formats.

#### 6.2.1.2 Analysis on ML model training

The term 'ML model training' has been defined differently by SA WG5, SA WG6 and RAN WG1, as illustrated in Table 6.2.1.2-1. RAN WG3 follows the definition of SA WG5.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.1.2-1: Definition of ML model training as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | ML model training |
| SA WG5 TS 28.105 [9] | A process performed by an ML training function to take training data, run it through an ML model algorithm, derive the associated loss and adjust the parameterization of that ML model iteratively based on the computed loss and generate the trained ML model. |
| SA WG6 TR 23.700-82 [7] | According to TS 28.105 [9], ML model training includes capabilities of an ML training function or service to take data, run it through an ML model, derive the associated loss and adjust the parameterization of that ML model based on the computed loss. |
| RAN WG1 TR 38.843 [3] | A process to train an AI/ML Model [by learning the input/output relationship] in a data driven manner and obtain the trained AI/ML Model for inference. |
| RAN WG3 TS 38.300 [11] | AI/ML Model Training follows the definition of the "ML model training" as specified in clause 3.1 of TS 28.105 [9]. |

The following unified definition for ‘ML model training’ is proposed:

**ML model training:** A process to train an ML Model by learning the input/output relationship in a data driven manner and obtain the trained ML Model for e.g. inference.

#### 6.2.1.3 Analysis on ML model re-training

The term 'ML model re-training' has been defined differently by SA WG5 and RAN WG1, as illustrated in Table 6.2.1.3-1. RAN WG1 introduces two new terms, i.e. ML model parameter update and ML model update, which is nothing but ML model re-training.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.1.3-1: Definition of ML model re-training / ML model parameter update as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | ML model re-training / ML model parameter update / ML model update |
| SA WG5 TS 28.105 [9] | *ML model re-training:* A process of training a previous version of an ML model and generate a new version. |
| RAN WG1 TR 38.843 [3] | *ML model parameter update:* A process of updating the model parameters of a model.  *Model update:* A process of updating the model parameters and/or model structure of a model |

The term ML model re-training is proposed to be adopted by all 3GPP Working Groups rather than using different terms such as ‘ML model parameter update’ or ‘ML model update’ which means exactly the same. The following unified definition for ‘ML model re-training’ is proposed:

**ML model re-training:** A process of training a previous version of an ML model and generate a new version.

#### 6.2.1.4 Analysis on ML model testing

The term 'ML model testing' has been defined differently by SA WG5 and RAN WG1, as illustrated in Table 6.2.1.4-1.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Editor's note: Further analysis of ML model monitoring, as per SA WG2 specifications, may be needed.

Table 6.2.1.4-1: Definition of ML model testing as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | ML model testing |
| SA WG5 TS 28.105 [9] | A process of testing an ML model using testing data. |
| RAN WG1 TR 38.843 [3] | A subprocess of training, to evaluate the performance of a final AI/ML model using a dataset different from one used for model training and validation. Differently from AI/ML model validation, testing does not assume subsequent tuning of the model. |

The following unified definition for ‘ML model testing’ is proposed:

**ML model testing:** A process of evaluating the performance of an ML model using test data different from data used for model training and validation.

#### 6.2.1.5 Analysis on ML model inference

The term 'ML model inference' has been defined differently by SA WG5, SA WG6 and RAN WG1, as illustrated in Table 6.2.1.5-1. RAN WG3 follows the definition of SA WG5.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.1.5-1: Definition of ML model inference as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | ML model inference |
| SA WG5 TS 28.105 [9] | A process of running a set of input data through a trained ML model to produce set of output data, such as predictions. |
| SA WG6 TR 23.700-82 [7] | According to TS 28.105 [9], ML model training includes capabilities of an ML model inference function that employs an ML model and/or AI decision entity to conduct inference. |
| RAN WG1 TR 38.843 [3] | A process of using a trained AI/ML model to produce a set of outputs based on a set of inputs. |
| RAN WG3 TS 38.300 [11] | AI/ML Model Inference follows the definition of the "AI/ML inference" as defined in clause 3.1 of TS 28.105 [9]. |

The following unified definition for ‘ML model inference’ is proposed:

**ML model inference:** A process of running a set of inputs through a trained ML model to produce a set of outputs.

#### 6.2.1.6 Analysis on ML model activation & ML model de-activation

The term 'ML model activation' and 'ML model deactivation' have been defined by RAN WG1, as illustrated in Table 6.2.1.6-1. SA WG5 mentions the terms ML activation and ML deactivation several times in TS 28.105 [9] but does not provide a definition.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.1.6-1: Definition of ML model inference as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | ML model activation & ML model de-activation |
| RAN WG1 TR 38.843 [3] | *ML Model activation:* enable an AI/ML model for a specific AI/ML-enabled feature.  *ML Model deactivation:* disable an AI/ML model for a specific AI/ML-enabled feature. |

The following unified definition for ‘ML model activation’ and ‘ML model deactivation’ is proposed:

**ML model activation:** A process to enable an ML model for inference.

**ML model deactivation:** A process to disable an ML model for inference.

#### 6.2.1.7 Analysis on ML model lifecycle

The term 'ML model lifecycle' has been defined by SA WG6, as illustrated in Table 6.2.1.7-1. However, SA WG2 TS 23.288 [8] SA WG2 TR 23.700-84 [4], SA WG4 TR 26.927 [12], SA WG5 TS 28.105 [9], SA WG6 TR 23.700-82 [7], RAN WG1 TR 38.843 [3] and RAN WG3 also mentions one or more phases of ML model life cycle without providing a clear definition of ML model lifecycle.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.1.7-1: Definition of ML model lifecycle as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | ML model lifecycle |
| SA WG6 TR 23.700-82 [7] | The lifecycle of an ML model includes data collection, data processing, model training, model verification, model, instantiation and deployment, model monitoring and termination of ML model components. |

The following unified definition for ‘ML model lifecycle’ is proposed:

**ML model lifecycle:** The end-to-end process typically consisting of data processing, model training, model testing, model deployment, model monitoring and model maintenance.NOTE 1: Data processing includes collecting and preparing the data for model training and model inference.

NOTE 2: Model training includes training and validating the model before deployment.

NOTE 3: Model testing includes testing the model before deployment.

NOTE 4: Model deployment includes making a trained ML model available for use in the target environment.

NOTE 5: Model monitoring includes observing the performance of the model once deployed for inference.

NOTE 6: Model maintenance includes updating the model, retraining the model and (de-)activating the model.

#### 6.2.1.8 Analysis on ML model lifecycle management

SA WG5 describes the ML model lifecycle in clause 4a.0 of TS 28.105 [9], and ML model lifecycle management capabilities for ML model training, ML model testing, ML inference emulation, ML model deployment and ML inference in clause 6.1 of TS 28.105 [9]. The terms ‘ML model-based lifecycle management’, ‘ML-enabled functionality’, and ‘Functionality-based lifecycle management’ have been defined by RAN1, as illustrated in Table 6.2.1.8-x.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

|  |  |
| --- | --- |
| **TSG (TS/TR)** | **ML model lifecycle management / Functionality-based lifecycle management** |
| 3GPP RAN1 TR 38.843 [e] | **ML model-based lifecycle management:** Operates based on identified logical models, where a model may be associated with specific configurations/conditions associated with UE capability of an AI/ML-enabled Feature / Feature Group and additional conditions (e.g., scenarios, sites, and datasets) as determined/identified between UE-side and NW-side. The models are identified at the Network, and Network/UE may activate/deactivate/select/switch individual AI/ML models via model ID.  **(ML-enabled) Functionality:** An AI/ML-enabled Feature/Feature Group enabled by configuration(s), where configuration(s) is(are) supported based on conditions indicated by UE capability.  **Functionality-based lifecycle management:** Signaling procedure where network indicates activation/deactivation/fallback/switching of AI/ML functionality via 3GPP signalling (e.g., RRC, MAC-CE, DCI); operates based on, at least, one configuration of AI/ML-enabled Feature/FG or specific configurations of an AI/ML-enabled Feature / Feature Group. |

**Table 6.2.1.8-x**: Definitions of ML model-based lifecycle management, ML-enabled functionality and Functionality-based lifecycle management as defined across 3GPP WGs.

The following unified definition for ‘ML model lifecycle management’ is proposed:

**ML model lifecycle management:** The management capabilities allowing a consumer to manage different phases of the ML model lifecycle as defined in clause 6.2.1.7.

The following definition for ‘Functionality-based lifecycle management’ is proposed for adoption by all 3GPP RAN Working Groups:

**Functionality-based lifecycle management:** Signaling procedure where network indicates activation/deactivation/fallback/switching of AI/ML functionality via 3GPP signalling (e.g., RRC, MAC-CE, DCI); operates based on, at least, one configuration of AI/ML-enabled Feature / Feature Group or specific configurations of an AI/ML-enabled Feature/FG.

NOTE 1: In the context of RAN1, RAN2 and RAN4, functionality-based lifecycle management does not consider training, testing and maintenance phases and consider them as implementation-specific.

NOTE 2: Applicability of Functionality-based lifecycle management definition to/in TSG SA WGs is optional.

### 6.2.2 Analysis on Horizontal Federated Learning

The term 'Horizontal Federated Learning' has been defined in SA WG2 and RAN WG1 defines 'Federated Learning', as illustrated in Table 6.2.2-1.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.2-1: Definition of Horizontal Federated Learning as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | Horizontal Federated Learning |
| SA WG2 TR 23.700-84 [4] | *Horizontal Federated Learning*: A federated learning technique without exchanging/sharing local data set, wherein the local data set in different FL clients for local model training have the same feature space for different samples (e.g. UE IDs). |
| RAN WG1 TR 38.843 [3] | *Federated Learning*: A machine learning technique that trains an AI/ML model across multiple decentralized edge nodes (e.g. UEs, gNBs) each performing local model training using local data samples. The technique requires multiple interactions of the model, but no exchange of local data samples. |

The following unified definition for ‘Horizontal Federated Learning’ is proposed:

**Horizontal Federated Learning:** A federated learning technique without exchanging/sharing local data set, wherein the local data set in different HFL clients for local model training have the same feature space for different samples.

### 6.2.3 Analysis on Vertical Federated Learning

The term 'Vertical Federated Learning' has been defined in SA WG2 and RAN WG1 defines 'Federated Learning', as illustrated in Table 6.2.3-1.

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

Table 6.2.3-1: Definition of Horizontal Federated Learning as defined across 3GPP WGs

|  |  |
| --- | --- |
| TSG (TS/TR) | Vertical Federated Learning |
| SA WG2 TR 23.700-84 [4] | *Vertical Federated Learning*: A federated learning technique without exchanging/sharing local data set, wherein the local data set in different VFL Participant for local model training have different feature spaces for the same samples (e.g. UE IDs). |
| RAN WG1 TR 38.843 [3] | *Federated Learning*: A machine learning technique that trains an AI/ML model across multiple decentralized edge nodes (e.g. UEs, gNBs) each performing local model training using local data samples. The technique requires multiple interactions of the model, but no exchange of local data samples. |

The following unified definition for ‘Vertical Federated Learning’ is proposed:

**Vertical Federated Learning:** A federated learning technique without exchanging/sharing local data set and local ML models, wherein the local data set in different VFL clients for local model training have different feature spaces for the same samples.

### 6.2.4 Analysis on Decision vs Prediction vs Output

RAN WG1 and RAN WG3 only uses "prediction" in all corresponding ML related TRs/TSs. SA WG2 uses "output" in all corresponding TRs/TSs where output may include both statistics and predictions. SA WG5 uses "decision" in all corresponding TRs/TSs with few occurrences of "prediction".

Editor's note: Further analysis may be needed, e.g. to determine whether a unified definition can be derived.

The term “**output**” is proposed as unified term since output may include decision or prediction or statistic or recommendation.

|  |
| --- |
| **End of modifications** |