**3GPP TSG- Meeting # *r05+***

**, , -**

|  |
| --- |
| *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 |  | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  |  |
|  |  |
| ***Source to WG:*** |  |
| ***Source to TSG:*** |  |
|  |  |
| ***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 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-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19) Rel-20 (Release 20)* |
|  |  |
| ***Reason for change:*** | There is an EN in clause 5.4 “Editor's note: Whether and how change of samples during a VFL training is supported is FFS”. For a Vertical Federated Learning operation with UEs as samples, it is a possibility that sample availabilty varies across training iterations. In some cases it may be possible to continue the training from a previously attained milestone rather than starting the training all new.  |
|  |  |
| ***Summary of change:*** | Removed EN from clause 5.4. In clause 6.2H.2.3.1: The intermediate training information from VFL Clients to VFL server includes the sample IDs that are missing compared to what was agreed in the preparation phase. This gives an indication to the VFL server that there has been a change in samples. The VFL server can also provide, as part of intermediate training information, training checkpoint information which is indicate whether VFL clients are required to store the state of models to be used as a fallback from a later iteration if a change in samples is detected. In clause 6.2H.3, added the checkpoint information in the intermediate model training results from VFL server to client and the details of indicating lost samples during a training iteration from the VFL client to server.  |
|  |  |
| ***Consequences if not approved:*** | Sample update during VFL training cannot be supported. |
|  |  |
| ***Clauses affected:*** | 5.4, 6.2H.2.3.1, 6.2H.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 <<<< |

## 5.4 Vertical Federated Learning (VFL)

Vertical Federated learning is a machine learning technique working without exchanging/sharing of local data set, while maintaining some level of coordination amongst VFL participants, when training and inference are performed on local ML Models, 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). Vertical Federated Learning may involve multiple NWDAFs and AFs.

For Vertical Federated Learning, there may be one NWDAF or one AF acting as a VFL server and one or multiple NWDAF(s) and/or one or multiple AF(s) acting as VFL Client(s). Vertical Federated Learning is available among NWDAFs or between NWDAF(s) and AF(s) within a single PLMN or between an AF and NWDAF(s) in a single PLMN.

The main functionalities of VFL server and VFL client include:

**VFL server:**

- An NWDAF or trusted AF acting as VFL server discovers and selects VFL client(s) (NWDAF(s) and/or AF(s)) to participate in a VFL procedure.

NOTE: When an untrusted AF is acting as VFL server, NEF discovers and selects candidate VFL client NWDAFs, then the AF determines final set of VFL clients.

- It requests VFL clients to do local ML model training for an Analytic ID, it assigns VFL correlation ID, and it requests to report intermediate results.

- It optionally locally trains ML Model with the available local data set.

- It combines intermediate results from VFL client(s) and VFL server and computes intermediate training results (e.g. gradient information, loss information) for updating its own local ML Model and the ML Models of VFL clients during the VFL training process and sends the intermediate training results towards VFL clients involved in the joint VFL training process. VFL server may send and receive separate message for each client.

NOTE: NEF forwards the message from/to NWDAF to/from untrusted AF.

- It determines to terminate the VFL training process.

- It stores VFL correlation ID and locally trained ML Model after VFL training process.

- It initiates the VFL inference process using VFL correlation ID.

- It combines local inference result from VFL clients and generates the final VFL inference result.

- It may send the final VFL inference result to the consumer.

- It supports to monitor the accuracy of the VFL model.

**VFL client:**

- It locally trains ML Model with the available local data set, which includes the data that may not be allowed to be shared with other VFL clients or VFL server due to e.g. data privacy, data security, data access rights.

- It computes the intermediate results for their local ML Models involved in the VFL training and provide reports with the intermediate results to the AF or NWDAF acting as VFL server.

- It stores VFL correlation ID and locally trained ML model after VFL training process.

- It performs inference based on the local model and local data and provides inference results to VFL server.

Editor's note: Details regarding Sample alignment and features alignment functionality or whether the functionality needs to be specified are FFS.

Editor's note: Accuracy monitoring in VFL is FFS.

Editor's note: For an NWDAF impacts of the split into AnLF and MTLF are FFS.

Vertical Federated Learning includes the following procedures:

- Registration of the NF profile including a list of VFL related information to NRF. Registration of the NWDAF profile to NRF is described in clause 5.2. Registration of the AF profile to NRF is described in clause 5.5. The procedure for registration and discovery of VFL server and VFL client is described in clause 6.2H.2.1.

- Preparation for VFL including sample alignment to ensure that all the VFL participants have common samples when training ML models as described in clause 6.2H.2.2.

- Training for VFL as described in clause 6.2H.2.3.

- Inference for VFL as described in clause 6.2H.2.4.

Editor's note: If additional functionality is specified in clause 6.2H, the reference to the corresponding clause will be added.

Editor's note: For an NWDAF impacts of the split into AnLF and MTLF are FFS.

|  |
| --- |
| >>>>>> SECOND CHANGE <<<< |

##### 6.2H.2.3.1 Training Procedure for Vertical Federated Learning when NWDAF is acting as VFL server

The figure 6.2H.2.3.1-1 below shows the training procedure for Vertical Federated Learning when NWDAF is acting as VFL server.

**

Figure 6.2H.2.3.1-1: Training procedure for Vertical Federated Learning when NWDAF is acting as VFL server

Editor's note: How the NEF assists the VFL training process as well as whether the service operations going via NEF is using the existing or new service operation are FFS.

Editor's note: The details of the services in the procedure and whether VFL Training Start Flag is needed are FFS.

Editor's note: It is FFS whether sample/feature information is required to be provided or updated in each training.

Editor's note: Whether and how to include interoperability information in the VFL training procedure is FFS.

Editor's note: Whether and how to define the trigger of VFL training is FFS.

Editor's note: Whether and how to transfer the confirmation in VFL Preparation Phase at the beginning of VFL Training Phase is FFS.

1. The NWDAF acting as VFL server determines the VFL clients that participate in VFL procedure in the VFL clients discovery and preparation phase as described in the clause 6.2H.2.1 and clause 6.2H.2.2.

NOTE 1: VFL Server can determine to start the training based on local configuration and agreement among vendors and/or application providers participating in the same group for specific VFL task(s).

Steps 2-6 are repeated until the training termination condition is reached.

2. To start the VFL training, the VFL server sends a request to start the training to all selected VFL clients The request includes VFL correlation ID, at least the parameters negotiated during the preparation phase, Optionally, the VFL Server includes: Analytic filter information, maximum response time (i.e. the maximum time between VFL clients receive intermediate model training information and send back intermediate training result).

Editor's note: Whether the parameters negotiated in the preparation phase are provided at the end of the preparation phase or at the start of the training is FFS.

 If the VFL procedure continues in subsequent iterations, the VFL server sends a request for a new VFL training iteration containing the VFL correlation ID and intermediate model training information to each of the VFL clients for next round of VFL training. The VFL Server may provide checkpoint information according to clause 6.2H.3.

2a. The VFL server sends a Nnwdaf\_VFLTraining\_Subscribe to the selected NWDAF VFL clients(s).

2b. The VFL server sends a Naf\_VFLTraining\_Subscribe to the selected trusted AF VFL clients(s).

2c. For each selected untrusted AF VFL clients, the VFL server sends a Nnef\_VFLTraining\_AFClient\_Subscribe to the NEF handling that AF.

2d. For each selected untrusted AF VFL clients, the NEF sends a Naf\_VFLTraining\_Subscribe to that AF. The NEF may also translate the analytic filter information if needed, e.g. TAIs into geographical area.

NOTE 2: In this release, the same NF associated with a VFL Server or VFL Client capability during the VFL training for a VFL correlation ID is also the same NF during the VFL inference.

Editor's note: Additional Parameters to be provided in the request are FFS.

Editor's note: It is FFS whether and how the local ML model is obtained by VFL Client in VFL training process.

3. [Optional] Each VFL client collects its local data by using the current mechanism if the VFL client has no local data already available. The data used by each VFL Client is collected as per alignment information.

4. During VFL training procedure, each VFL client further trains the local ML model associated with the same VFL Correlation ID based on their own collected or available data and when applicable (e.g. after the first round of training) and possible intermediate model training information distributed by the VFL server in the previous training iteration. Each VFL Client computes and reports the client intermediate training result of the local ML model to the VFL server. VFL client(s) may also report a sample loss according to clause 6.2H.3NOTE 3: The intermediate model training information and intermediate training result are constructed in per sample granularity.

Editor's note: It is FFS and may depend on the service design: When the clients report the client intermediate training result, it also includes the corresponding VFL correlation ID.

Editor's note: Whether and how to exchange intermediate training result among VFL Clients is FFS.

5. Each VFL client reports the computed client intermediate training result of the local ML model to the VFL server.

5a. A NWDAF VFL client sends a Nnwdaf\_VFLTraining\_Notify.

5b. A trusted AF VFL client sends a Naf\_VFLTraining\_Notify to the VFL server.

5c. An untrusted AF VFL client sends a Naf\_VFLTraining\_Notify to the NEF.

5d. For each untrusted AF VFL client , the NEF converts any external identifiers to internal identifiers and sends a Nnef\_VFLTraining\_AFClient\_Notify to the VFL server.

6. The VFL server may collect the local data and generate its own local intermediate training result. The NWDAF acting as VFL Server computes the intermediate model training information (e.g. gradient information or loss information) based on the VFL Client(s) intermediate training result(s) received in step 4, its own local intermediate results and the label. The intermediate model training information is used for updating the models of VFL clients. Different intermediate model training information may be computed for different VFL clients and for the VFL Server itself.

 The VFL server may also compute the ML model metric (e.g. ML model accuracy) based on all the intermediate training result received from VFL clients and the label.

Editor's note: Whether weight of the VFL Client is computed by VFL server is FFS.

Editor's note: Whether VFL server and VFL clients share feature information is FFS.

7. [Optional] The NWDAF acting as VFL server evaluates (e.g. based on the convergence of a loss function or loss value and/or if the pre-set iteration number is reached) whether VFL Training process converged. If the VFL Server evaluates the VFL Training process did not converge, the NWDAF acting as a VFL Server determines another round of VFL training is required and repeats step 2 - 6. If the VFL Server evaluates the VFL training process converged, it determines the VFL Training is completed. In this case, the VFL Server terminates the current VFL training process via step 7.

 The VFL training termination decision may be also made as follows:

 Based on the consumer request, the VFL server sends VFL status report to the consumer. The status report may include model metric (e.g. ML model accuracy).

Editor's note: The content of the VFL status report is FFS.

Editor's note: Whether VFL server sending convergence report to the VFL client and what is convergence report are FFS.

 The consumer decides whether the current model can fulfil the requirement, e.g. ML model metric is satisfactory for the consumer and determines to stop or continue the training process. The consumer continues the training process or stops the training process.

 Based on the subscription request sent from the consumer, the VFL server updates or terminates the current VFL training process.

Editor's note: Whether the ML model metric (e.g. ML model accuracy) defined for HFL can be re-applied to VFL is FFS.

8. The VFL server sends VFL training termination message to VFL Client if it decides to terminate the VFL training process, the termination message contains VFL Correlation ID.

8a. The VFL server sends a Nnwdaf\_VFLTraining\_Unsubscribe t to the selected NWDAF VFL clients(s).

8b. The VFL server sends a Naf\_VFLTraining\_Unsubscribe to the selected trusted AF VFL clients(s).

8c. For each selected untrusted AF VFL clients, the VFL server sends a Nnef\_VFLTraining\_AFClient\_Unubscribe to the NEF handling that AF.

8d. For each selected untrusted AF VFL clients, the NEF sends a Naf\_VFLTraining\_Unsubscribe to that AF.

9. The VFL Server, stores VFL correlation ID, the local trained ML Model, the mapping information of the VFL correlation ID to the following parameters: Analytics ID related to the VFL training process, locally trained Model. Additionally, the VFL server stores the VFL client information, which may be used to determine associated VFL client in the VFL inference.

 Each VFL client stores VFL correlation ID, the locally trained ML Model, the mapping information of the VFL correlation ID to locally trained Model

NOTE 4: The VFL correlation ID and the stored mapping information are used later for inference as described in Clause 6.2H.2.4.1.

Editor's note: Whether VFL Training termination Flag in the termination message is required is determined after settling down the service operation.

NOTE 5: If untrusted AF is involved in VFL Clients, the message between NWDAF acting as VFL Server and the untrusted AF is via NEF.

|  |
| --- |
| >>>>>> THIRD CHANGE <<<< |

### 6.2H.3 Contents of ML Model Training service for Vertical Federated Learning

The consumers of the ML Model training services may provide the input parameters in Nnwdaf\_VFLTraining service or Naf\_VFLTraining service or in Nnefe\_VFLTraining service as listed below:

- Analytics ID: identifies the analytics for which the ML Model is requested to be trained.

- VFL Interoperability Information that indicates the intermediate results that the VFL Server supports (e.g. activation function, gradients, type of loss), the content of the VFL Interoperability Information is not standardized in this release.

- In the preparation phase, the initial list of samples selected by VFL Server.

- (Optional) In later steps in training, the VFL server can indicate sample loss from the list sent in first step of training procedure to indicate which samples aligned in the preparation phase will not be part of the rest of training procedure.

- (Optional), VFL server may indicate model status has to be saved as a checkpoint or if the training has to restart from the model status of a previous checkpoint (e.g. training iteration number).

The VFL client provides to the consumer of the ML Model training service operations the output information in as listed below:

- VFL Interoperability information supported by each VFL Clients.

- In the preparation phase, the list of samples accepted by VFL client.

- (Optional) In training the VFL client may indicate sample loss from the list received in first step of training procedure to indicate which samples were lost and will not be part of the rest of training procedure.

- The Feature ID(s) that the VFL client supports. A Feature ID indicates that what features the VFL client can use for an Analytics ID, the values of the Feature ID are not standardized.

Editor's note: Which parameters are optional, or mandatory is FFS, the list of parameters also needs to be extended.

|  |
| --- |
| >>>>>> END OF CHANGES <<<< |